Technical Memorandum No. 108

A Minicomputer Application

by

KURT METZGER

D D C
RECEIVED
SEP 29 1972
RECEIVED
D

COOLEY ELECTRONICS LABORATORY

Department of Electrical and Computer Engineering
The University of Michigan
Ann Arbor, Michigan

Approved for public release; distribution unlimited.

September 1972

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)	2a. REPORT SECURITY CLASSIFICATION
Cooley Electronics Laboratory The University of Michigan Ann Arbor, Michigan 48105	Unclassified
	2b. GROUP

3. REPORT TITLE

A MINICOMPUTER APPLICATION

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Memorandum No. 108 -- April 1967 through July 1971

5. AUTHOR(S) (First name, middle initial, last name)

Kurt Metzger

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
September 1972	52	

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
N00014-67-A-0181-0032	TM 108
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	036040-15-M

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Office of Naval Research Department of the Navy Arlington, Virginia 22217

13. ABSTRACT

The Acoustic Signal Processing Group at Cooley Electronics Laboratory, The University of Michigan, has used minicomputers in its research efforts since early 1967. This memorandum describes the group's activities in converting its minicomputers into viable data acquisition/processing systems. The period covered is from April 1967 through July 1971.

DD FORM 1473 (1 NOV 65)

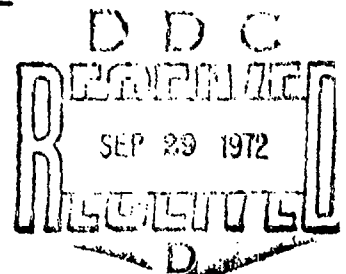| 14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Minicomputers						
LINC-8						
FIELD-8						
PDP-8/L						
Acoustic signal processing						
MIMI						
Software						
Hardware						
LINC tape						

Technical Memorandum No. 108

036040-15-M

A MINICOMPUTER APPLICATION

by

Kurt Metzger

Approved by: _Theodore G. Birdsall_
Theodore G. Birdsall

COOLEY ELECTRONICS LABORATORY

Department of Electrical and Computer Engineering
The University of Michigan
Ann Arbor, Michigan

for

September 1972

FOREWORD

This memorandum is a reprint of a recent talk given by the author. It is hoped that the experiences described here will be of interest and perhaps of some use to other minicomputer users.

As with all group efforts, many people have contributed to the overall effort. Professor T. G. Birdsall was the project director, and he established the guidelines for the project. Of particular note are the contributions of Jerry Cederquist, Brian Barton and Wayne Von Wald of Cooley Electronics Laboratory.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1. INTRODUCTION

The intent of this talk is to share with you the experiences of a small research group at The University of Michigan in its efforts to use the minicomputer as a research tool. Although the application described here is primarily a laboratory application, it is felt that the problems encountered and the approaches taken to solve them are of general interest.

The Signal Processing Group of the U of M's Cooley Electronics Laboratory (CEL), under the direction of Professor T. G. Birdsall, is involved in a cooperative effort with the University of Miami's Rosenstiel School of Marine and Atmospheric Sciences (RSMAS) in studying the characteristics of fixed site acoustic propagation in the ocean. The joint effort has been nicknamed MIMI (Miami-Michigan) and is funded by the Office of Naval Research, Code 468.

Project MIMI has been using minicomputers since 1967 and currently owns three LINC-8 computers and one PDP-8/L. Three dual processor PDP-8/e field packages are being purchased.

2. BACKGROUND

Project MIMI, the cooperative effort between RSMAS and CEL, was started in the summer of 1963. The purpose of this effort was, and still is, to study and to relate to environmental conditions the propagation of acoustic signals transmitted across the Straits of Florida from Miami on the mainland to Bimini Island in the Bahamas.

The group at RSMAS has the responsibility of operating and maintaining the site, of running long term studies, and of relating propagation data to environmental parameters, such as tide height, water temperature, etc. The group at CEL provides support to RSMAS in suggesting signal processing schemes and in developing new experiments.

From 1963 through April of 1967 only two approaches to signal processing were available to MIMI. The approach at RSMAS was to design and build analog equipment tailored to the desired experiment. Building and testing this equipment was slow and tedious and resulted in experimental setups that could not be easily modified. While RSMAS used specialized hardware to run long term studies, the group at CEL required data which could be used in running computer experiments. The results of these computer experiments were used to evaluate the potential of various experiments and to establish

2

designs for new hardware. Analog tape recordings were used to supply CEL with the required data. The analog tapes were digitized onto IBM tape using a special A/D conversion facility possessed by CEL.

The use of analog tape recordings presented CEL with several problems. Our signal processing techniques depend upon the processing equipment maintaining perfect time synchronism with the actual transmission. The required timing information was recorded on its own tape track at the same time that the analog data was being recorded. Unfortunately, small imperfections in the tape often caused the loss of timing information thus limiting the effectiveness of the processing techniques. Also, problems were encountered in digitizing the data onto IBM tapes in long enough segments to be useful. It was not uncommon for CEL to write a 2400-ft IBM tape as one record, containing only 15 minutes of data. These tapes were read into the U of M's 7090 using special programming techniques. This was a difficult and expensive procedure.

Since the CEL group was already using the computer in its experimental investigations of signal processing techniques, it was natural to consider processing the reception at IMS using a digital computer. The advantage of this approach was that one piece of equipment, the computer, could be used to implement any number of processing techniques. The problem at that time (1966) was to find a reasonably priced computer with sufficient capabilities.

In late 1966 the group at RSMAS purchased two Digital Equipment Corporation (DEC) LINC-8 computers for use in running signal processing experiments. One machine was to be kept at RSMAS's site on Virginia Key, Florida, and was to be used for program development and for processing the outputs of local hydrophones. The other machine was to be placed on Bimini and used as the 40-mile receiver. The Bimini machine was sent to CEL first for programming and then later was to be sent on to Bimini. The two machines were delivered in April of 1967, one to RSMAS and the other to CEL.

The LINC-8, which consists of two processing units sharing a common memory, was the result of a move by DEC to get into the laboratory instrument computer market by combining a popular medical computer developed by NIH with their own PDP-8.

The following few paragraphs describe the LINC-8 and are from DEC's 1966 Small Computer Handbook.

> Both the LINC and the PDP-8 are one-address, fixed-word length, parallel computers using 12-bit binary arithmetic. Cycle time of the 4096-word random-address magnetic-core memory, which is shared by both computers, is 1.5 μsec. Standard features of the LINC-8 include indexed and indirect addressing, facilities for sensing a wide variety of external signals, and program interruption or pausing as functions of input/output device conditions.

> The LINC-8 system combines two major subsystems, the LINC subsystem operating as an adjunct to the PDP-8 subsystem, and provides two distinct modes of operation. In the LINC mode, the PDP-8 operates primarily as a supporting system component supplying

special interpretive routines for some of the more complex LINC instructions. In the PDP-8 mode, the LINC subsystem is disabled, and all of the standard features of the PDP-8 augmented by the dual magnetic tape transport, are operable.

The LINC subsystem (referred to simply as the LINC) combines a processor, input/output equipment, and operator console, and uses part of the basic 4096-word PDP-8 memory for storage of LINC programs and data. In the LINC mode of operation, instructions and data are retrieved from or stored in the memory under control of the data break facilities of the PDP-8 and are interpreted and processed by the LINC processor. The memory is continuously cycling, automatically performing a read and write operation during each computer cycle. Certain complex LINC instructions, namely the magnetic tape and operate instructions, together with the sensing and control of the LINC operator console, are executed by interpretive PDP-8 routines also residing in basic memory. The program interrupt facilities of the PDP-8 call these routines into operation, and the LINC is disabled during their execution.

Among the peripheral I/O devices found on the LINC are:

a CRT display, a 16 channel A/D converter, a relay register, sense lines, sense switches and dual magnetic tapes (non-industry standard).

When the decision to buy the LINC-8's was made, the LINC-8 and the PDP-8 were essentially the only "minicomputers" on the market. The basic LINC-8 as described above carried a price tag of $ 38,500 but even at this price it was less expensive than a comparably equipped PDP-8.

The key factors influencing the purchase were:

Eight A/D channels were available for use with external analog inputs.

The scope permitted displays to be generated in real time allowing continuous monitoring of signal processing results.

The magnetic tapes, although not IBM compatible, permitted data and programs to be stored at fixed predetermined locations on tape. This capability provided the user with the capability to replace and update information contained in specified sections of the tape without disturbing any other prerecorded information. That is, the tapes provided random access similar to that of a drum or a disc rather than sequential access like that found on IBM tape. This feature proved very important in later software development at CEL.

Extensive support software was available. DEC supported the PDP-8, and Washington University in St. Louis, was funded by the National Institutes of Health to support the LINC.

The arrival of the Bimini LINC-8 at CEL triggered a marathon round of programming sessions. The staff had a new tool (or toy) and the best way to learn how to use it effectively was to use it. Our first efforts consisted of "demonstration" programs, that is, programs for speech processing, studying the properties of pseudo-random numbers and statistical studies.

By the summer of 1967, the "Bimini" LINC-8 had made such an impression at CEL that it was decided to purchase a LINC-8 for CEL. In addition to the basic machine it was decided to order an additional 4096 words of memory ($10,000). The additional memory was purchased for the following reasons:

The PDP-8 processor in the LINC-8 was start-
ing to be used more frequently. However, DEC
did not have any "system" software designed for
use on the PDP-8 side of the LINC-8 (their 8-
LIBRARY SYSTEM was woefully inadequate).
It was felt that, using the extra 4K, an effec-
tive system could be produced using much of
DEC's current 4K software.

There was a rumor that DEC was about to re-
lease an 8K FORTRAN-II package.

The extra memory would be useful in develop-
ing large signal processing programs.

The CEL LINC-8 arrived in November of 1967. The Bimini

LINC-8 was supposed to leave CEL that same month; however, prob-

lems at RSMAS delayed the actual departure until Novembeber of 1968.

Thus, for a one-year period CEL had two minicomputers.

In spite of the apparent abundance of computer power, com-

petition for machine use was fierce. This competition eventually

resulted in the use of a sign-up sheet and a user priority system.

Each user was allowed two hours on the machine at an assigned

priority level on a first-served basis. After two hours his priority

dropped to the lowest level. A user could only be bumped on the

hour. Thus, if a user could get on, he was guaranteed one hour on

a computer regardless of his priority. Priorities were re-estab-

lished at 7:30 a. m. and 6:30 p. m. The period from 12:00 a. m. to

7:30 a. m. was unassigned.

In the period from November 1967 through November 1968 several signal processing programs were written and various experimental packages were developed. In addition to this effort, two simple programming systems based on the 8-LIBRARY SYSTEM were developed for use with the PDP-8 processor. The first of these was an attempt to automate computer use and minimize the use of paper tape in DEC's paper tape "system." The second system was devoted to implementing a paper-tapeless 8K FORTRAN operating system.

The loss of the Bimini LINC-8 brought on a minor crisis at CEL. We lost almost half of our computer power. Jerry Cederquist of CEL undertook a software development program in an attempt to recoup the loss in hardware through an improvement in operating software. The finished result was unveiled in March of 1969 as the Cooley Programming System (CPS).

CPS is an 8K PDP-8 operating system externally modeled after the U of M's IBM 360 time-sharing system, MTS (Michigan Terminal System). CPS performed as advertised and by May of 1969 the LINC processor in the CEL LINC-8 was, for all practical purposes, dead.

June and July of 1969 saw the development of 4K-CPS for use on RSMAS's 4K machines. Also, work was started on a small transportable computer system based on DEC's PDP-8/L for use

in field experiments. The system was dubbed the "FIELD-8."

In putting together the FIELD-8 system we attempted to duplicate or extend those features that had proven most useful on the LINC-8. The single most important de   e on the LINC-8 was the LINC tape system. The LINC tapes not only provided a convenient storage medium for data (230,000 PDP-8 words per $8 tape) but, because of CPS, were the bases for all of our programming. Because of its importance and the desire to allow trading tapes between computers we built our own LINC tape controller for use on the PDP-8/L. The FIELD-8 became operational in the early part of 1970 and has proved to be very reliable.

In 1970 project MIMI undertook three field trips with its FIELD-8 system. The first of these was in July to Bimini in the Bahamas. This trip required shipping the computer by truck and seaplane.   The project fielded eight people from Michigan for this series of experiments. Three major experiments were conducted, each one requiring 8K words of memory. Several lesser experiments were also run. Programming techniques used ranged from simple flag test I/O to foreground-background I/O and multiprogramming. Two significant conclusions of these tests were: (1) we had too many people in the field, (2) we always had the wrong experiment running at the right time (i.e., no matter what program was running we always wondered how changing conditions would have

affected both of the other programs).

By September of 1970 we had changed our program requirements so that all three programs could run simultaneously using 8K words of memory. A two-week field trip was made to the Navy's Underwater Sound Laboratory in New London, Connecticut. Two people made this trip in a station wagon.

November of 1970 saw the return to Bimini of the FIELD-8 and one man for a three-week field experiment.

April of 1971 saw work start on the design and acquisition of a new, extended capability field package. The resulting system was based on DEC's PDP-8/e computer using dual processors. CEL purchased one system and the Miami group, two. It is anticipated that this system will be able to handle our data gathering and processing needs for the next several years. In order to maintain compatibility with our other machines it was again decided to design our own program-compatible LINC tape controller.

3. HARDWARE

3.1 LINC-8

The CEL LINC-8 configuration as it currently exists is shown in Fig. 1. To the LINC-8, we have added an interface to our old IBM tape equipment and a data set. We have also made several internal modifications to the LINC-8 in order to remedy what we considered to be design oversights.

The most serious flaw in the LINC-8 architecture, from our viewpoint (and one which we have not fixed), was the use of the same internal registers for sampling and for data transfers to and from magnetic tape. This conflict in register use prevents us from sampling input data during the time that we are dumping processed results onto tape. So far we have managed to program around this, but with some difficulty. This problem was discovered only upon reading the maintenance manual sent with the machine.

3.1.1 IBM Tape Interface.
The IBM tape interface shown in Fig. 2 is in reality between the LINC-8 and CEL's old digitizing equipment, which includes an IBM tape drive controller. It was constructed using the "spare" modules used in the digitizing equipment. The ability to read and write IBM tapes was intended to permit the transfer of large amounts of data between the LINC-8 and the U of M's
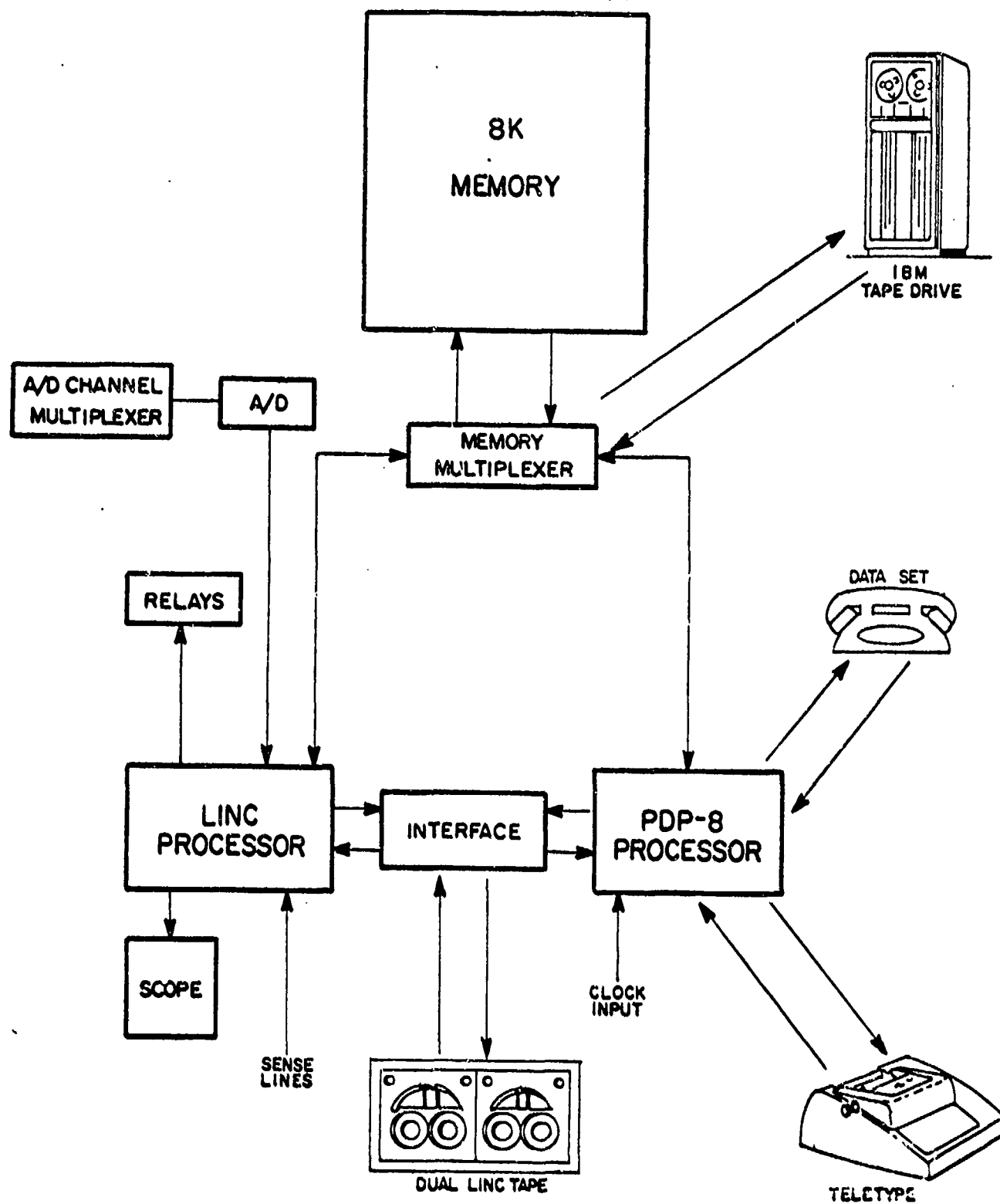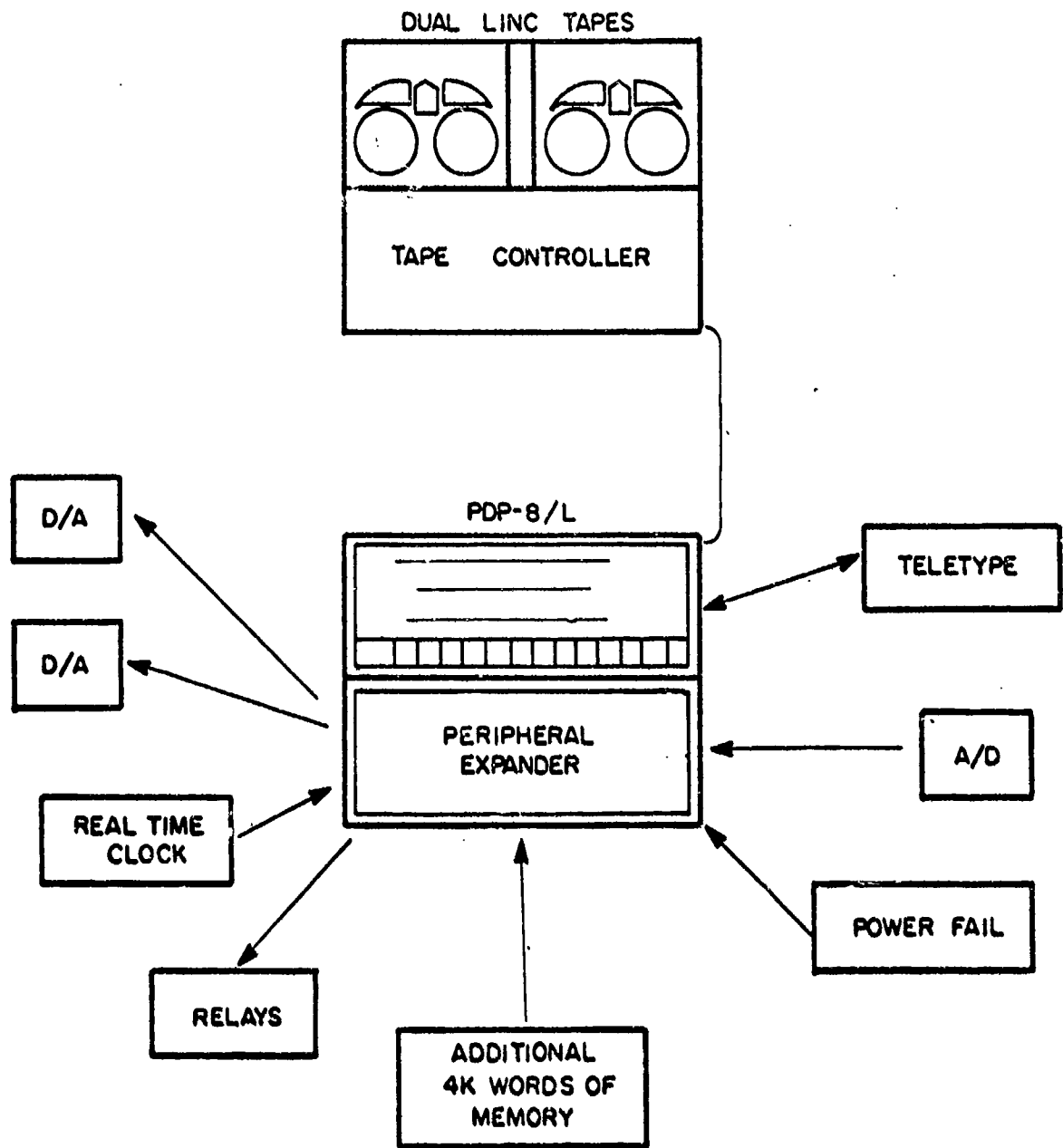
Fig. 1. CEL LINC-8

Fig. 2. CEL FIELD-8

IBM 360.

Initially it was felt that the interface would find extensive use. However, since its completion in May of 1968, it has been used only four times and only once to send data to the "360." The reasons for this seem to be:

1. Software developments have made it possible to use the LINC-8 to do many jobs that we had thought could only be done on the 360.

2. General purpose data conversion programs for use on the 360 were never completely developed.

3. The interface was awkward to use. Since it was not in constant use, it kept getting taken apart.

The programming required in the PDP-8 to support the interface was minimal, only three instructions were required to effect a data transfer. Data transfers were made using the cycle stealing direct memory access (data-break) feature of the PDP-8.

3.1.2 Data Set. The data set, which we connected to the LINC-8, is Bell Telephone's 113A data phone. This is a low-speed (10-30 characters per second) data set and is usually found connecting remote terminals to time-sharing systems through the standard telephone network. The 113A was added to the LINC-8 for two reasons. First, it provided us with another link to the U of M's 360, and, second, it allowed the use of remote terminals in place of the LINC-8's standard ASR-33 Teletype.

The data set has had some use in "talking" to MTS, but not

an extensive amount. This is again due in part to the LINC-8 being

able to take over many of the tasks which were to have been given

to the 360. The data set has had extensive use in allowing remote

operation of the LINC-8.

Recently the LINC-8's ASR-33 was supplemented by a new

high-speed terminal, a Memorex 1240. The LINC-8 was modified

to handle data rates of 10, 30, 60 and 120 characters per second.

The use of the Memorex speeded up the processes of generating

program listings by a factor of 6. Thus, what were one-hour listings

(only too common) now are only ten-minute listings, a much appreci-

ated improvement.

3.2 FIELD-8

Figure 2 shows the components involved used in our transport-

able data acquisition system, the FIELD-8. The design of this piece

of equipment was started 18 months after the arrival of the CEL

LINC-8. By then, for our applications, it had been decided that the

PDP-8 was the more powerful of the two processors in the LINC-8.

By this time the only LINC features that we were using were the

magnetic tape unit and the A/D converter.

Work on the FIELD-8 was started about May of 1969. The

FIELD-8 was to be a small, portable, computerized signal proces-

sing system intended to be operated in the field at sites other than

RSMAS's in Miami. (The Bimini LINC-8 never made it to Bimini.)

The PDP-8/L was selected to be the heart of the system because it was small (70 lbs), inexpensive ($8,500), and program compatible with MIMI's three LINC-8 computers.

After much soul-searching, the CEL FIELD-8 shown in Fig. 2 was finally settled on.

3.2.1 DEC Peripherals. The additional 4K words of memory, power-fail restart option, and real-time clock shown in Fig. 2 were purchased with the computer.

The extra 4K words of memory were included because of the additional processing capability it would provide. Since we had decided on supplying a LINC tape interface, it also permitted the use of our CPS software out in the field.

The power-fail restart option was added so that the computer could be left unattended for long periods at sites with unreliable power. Programs were to be written so that if the power failed, then when it came back on, the computer could restart itself and continue processing, without human intervention.

The real-time clock option was included in order to provide an internal timekeeping ability. A time-of-day clock would have been more desirable but DEC does not make one, and we had our hands full building the LINC tape controller. This feature is planned to

be included on our future machines.

3.2.2 CEL Peripherals.
The above three DEC supplied options came installed in a cabinet called a peripheral expanded unit (BA08). The BA08 comes prewired for almost all standard options which DEC sells for its machines. However, since we had no intention of ever adding any of these options, we wanted to use the available space for our own peripheral devices. In order to do this we had to spend about one man-week removing the useless wires and getting the BA08 working again.

Using modules purchased from DEC we then installed three D/A converters and an A/D converter (single channel). Three D/A outputs were provided in order to allow the generation of X-Y displays on an external scope and to drive chart recorders, etc. The capabilities of the A/D converter will probably be upgraded to two or four channels.

A set of six relays under computer control were also furnished in order to provide the FIELD-8 with the same control capability possessed by the LINC-8.

3.2.3 Tape System.
When we designed the FIELD-8, we decided that we were not willing to give up in the field any of the programming capabilities that we had in the lab. In addition, we did not want to have to maintain two separate software systems, one for the LINC-8 and one for the FIELD-8. The only way to achieve

this equality-compatibility goal was to use an 8K PDP-8/L and supply a LINC tape system.

DEC was contacted about the possibility of providing a LINC tape system for the PDP-8/L. Unfortunately the prices quoted were beyond our ability to pay (single unit economics are usually prohibitively expensive).

Having a working model of a LINC tape controller in our LINC-8, we investigated the possibility of doing the job ourselves. Not counting labor, we managed the job, using DEC supplied modules and a dual LINC tape drive, for about $6,500. It took about two man-months to design and build the unit. Because we copied the interface structure as found in the LINC-8 it is possible to use all our system software directly on the FIELD-8 without modification. In fact, DEC's 8-LIBRARY SYSTEM also runs on the FIELD-8.

The price paid for this convenience was two-fold. First, we had to do the job ourselves; second, we were stuck with a tape system which was extremely awkward to program, i.e., the original design wasn't too hot. The controller has been in use for quite some time now and has proven to be quite reliable. It is designed to be plugged into the I/O bus of any unmodified PDP-8/L computer. When the work on the FIELD-8 was started, a comparable DEC tape system would have cost about $13,000. (DEC tapes and LINC tapes

are not compatible.) Since this time, DEC has announced new models and the present cost is on the order of seven or eight thousand dollars. Unless there is some special need, the approach taken by CEL should not be copied.

3.3 New Field Computer System

By May 1971 we realized that project MIMI's projected experimental program would require three or four additional field packages. After much research and discussion, a dual processor PDP-8/e configuration was settled on as the best compromise selection. The main considerations were:

(1) time required to become operational

(2) transfer of current programs and skills

(3) price

(4) compatibility with the four existing MIMI machines

A new tape controller was designed by CEL for use on the PDP-8/e in order to be able to read and write LINC tapes. The tape control instruction set again mimics that found in the LINC-8. The controller itself is designed to fit onto a board which mounts directly on the tape drive. The unit was designed to interface to the positive external I/O bus. It can operate on any PDP-8 computer which is so equipped. The cost is about $6200.

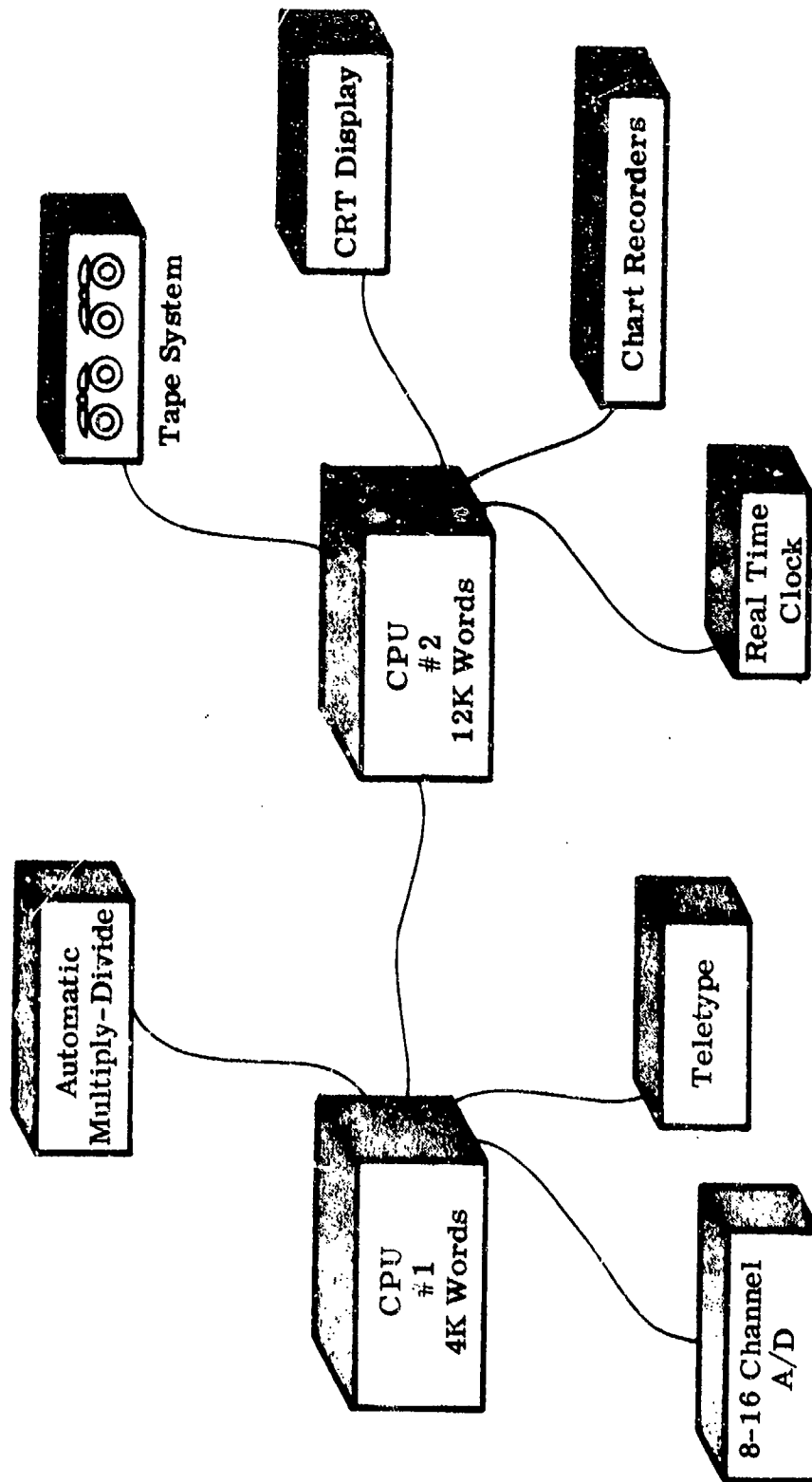The dual processor configuration is illustrated in Fig. 3.

Fig. 3. Dual processor configuration

With the exception of the tape controller all of the components were purchased from the manufacturer. One of the design goals was to minimize the number of specials and modifications required to implement the system.

4. SOFTWARE

4.1 Manufacturer Supplied Software

Along with the LINC-8, CEL received DEC's standard LINC and PDP-8 programming packages. Figure 4 summarizes the main programming packages which were provided. Since we purchased the LINC-8, several other components have become available.

4.1.1 LINC Software.

Because Washington University was being funded by the Government to develop and maintain the software for the LINC computers (of which the LINC-8 was one of four or five different makes), DEC did very little of its own software development for the LINC-8. However, it did establish and maintain an active LINC-8 division in its DEC Users Society (DECUS).

The most important piece of software received for use in the LINC processor was M. A. Wilkes' LAP-6 operating system. The LAP-6 system consists of an interactive text editor (using the LINC display scope), a file system and a program execution system. The system is completely contained on magnetic tape and does not use any paper tape. Most of the system's operation is centered around the text editor where "meta" commands are used to control most of the system's operation. LAP-6 is a notable achievement in that it was designed to operate in a 2048 word machine and, for most of its operations, only requires one tape unit.

22

LINC: LAP-4 and LAP-6 (Text editor, assembler and file

system)

Utilities | arithmetic

symbol manipulation

(The LINC software was furnished by Washington Uni-

versity, St. Louis, Missouri.)

PDP-8:   8-LIBRARY SYSTEM (file system)

Text Editor

PAL III } Assemblers
MACRO-8

SABR
8K-FORTRAN } 8K-FORTRAN Package
Reloc. Ldr

FOCAL

Numercial and
I/O Utilities

Floating Point
Interpreter

Fig. 4.   Software supplied by DEC

CEL made extensive use of LAP-6 to develop LINC oriented signal processing programs. However, with the advent of improved software for the PDP-8 side of the LINC-8, LAP-6 has fallen into disuse. In addition to LAP-6, we received several utility and demonstrator programs. Few of these have been put to practical use at CEL.

4.1.2 PDP-8 Software. The PDP-8 was (and still is) one of DEC's bread and butter machines. Thus, the software provided by DEC for use in the PDP-8 side of the LINC-8 was relatively extensive. In addition to the standard PDP-8 software, DEC also provided a simple LINC tape based PDP-8 filing system, called the 8-LIBRARY SYSTEM, for use on the LINC-8.

Using the 8-LIBRARY SYSTEM the user could save programs onto LINC tape and later call them back into core. Other than SAVE, DELETE and RUN no other services were provided. However, it did permit loading DEC's paper tape system components (text editor, assembler, etc.) into core from LINC tape rather than from paper tape.

The rest of the furnished software made up DEC's paper tape operating system and was designed for use on the basic 4K PDP-8 requiring only the ASR-33 Teletype.

The following steps were necessary in order to use the paper tape system in preparing program for the PDP-8. (The details are

being presented here because on most minicomputers this is how
it is done!)

(1) The text editor is loaded.
(2) A program is typed in and checked.
(3) The finished program is punched onto paper tape.
(4) The assembler is loaded.
(5) The source program paper tape is passed through the assembler (pass 1).
(6) The source program paper tape is passed through the assembler (pass 2) and the binary version of the source program is punched onto paper tape.
(7) The source program paper tape is passed through the assembler (pass 3) and a symbolic listing is produced.
(8) The binary paper tape is loaded.
(9) Debugging starts.

We were fortunate in that we could use the 8-LIBRARY SYSTEM
to load the editor and assembler. It could also be used to hold the
text editor containing the source program. If a program had to be
modified, it was not necessary to reload the source paper tape back
into the text editor. Using this software, approximately one-half hour
was required to assemble small programs of about 100 instructions.
If we had to load the editor and assembler using paper tape, then the
required time would have just about doubled. Of course, the addition
of high-speed paper tape to the LINC-8 would have reduced the amount
of time required.

Unfortunately the 8-LIBRARY SYSTEM was strictly a 4K sys-
tem. There was no convenient way to incorporate DEC's 8K FORTRAN
system. In effect it wasted our extra memory capacity.

4.2 CEL Developed Software

The most basic and important software developments at CEL
were:

(1) Learning how to program the LINC tapes.
(2) Learning how to run under interrupts.
(3) Writing our own operating system, CPS.
(4) Multiprogramming our signal processing programs.

The development of LINC tape routines for use in the PDP-8
side of the LINC-8 provided the PDP-8 programmer the same access
to LINC tapes enjoyed by the LINC programmers. In fact, since the
LINC tapes were controlled from the PDP-8, the PDP-8 user actually
could make more effective use of the LINC tapes than could the LINC
user.

The use of the interrupt facility in the PDP-8 has permitted
the development of complex signal processing programs. Early at-
tempts at writing processing programs not using interrupts were
of limited success because efficient use was not being made of the
available computer time. Operating under interrupts has only slight-
ly complicated program design and has paid off handsomely in increased
capabilities.

As far as operating systems go, we felt that the PDP-8 pro-
cessor was superior to the LINC processor, but we did not have any
adequate software to exploit it. Two earlier attempts had been made
at modifying DEC's 8-LIBRARY SYSTEM. However, these turned out

to be half measures and really did not effectively use the LINC tapes.

Thus, work was started, in November 1968, on a part-time basis, to

develop an operating system that would:

(1) eliminate paper tape
(2) automate program filing
(3) effectively make use of 8K words of memory, and
(4) coherently integrate much of DEC's existing software.

The desired system was completed in March of 1969 and was called

the Cooley Programming System (CPS).

Essentially, the system was composed of three functional

units: a command interpreter, a file system and a program execu-

tion system. All "system" functions such as text editing, assembling

and compiling were performed, not by the system, but by programs

contained in the system. Figure 5 lists the main programs found in

CPS.

An appreciable speed increase in programming was obtained

using CPS. For example, only 2-1/2 minutes were required to

assemble the assembler itself. Compare this against the half-hour

that used to be required for small programs.

In July of 1969 a version of CPS, 4K-CPS, was written for use

on the basic 4K LINC-8. This system was developed in order to

provide RSMAS with an improved programming capability and to

facilitate the transfer of programs between RSMAS and CEL.

Command Interpreter

Program Execution System   Control Program

File System

*ED	Symbolic text editor
*ASM *ALOAD	Absolute assembler and core image generator
*SABR *FORTRAN *RLOAD	8K-FORTRAN system components
*TAPMARK *TAPCOPY *TRW2	Magnetic tape utilities
*PUNCH *LIST	Symbolic output utilities
*PAPERBIN	Binary input utility
*FOCAL	Interactive user oriented language

Fig. 5. CPS (Cooley Programming System)

CPS was eventually sold to DEC and motivated them to develop a similar system for use on 8K and larger PDP-8's using "standard" DEC peripherals (i.e., not LINC tape).

In addition to the utility programs received from DEC, we have found it necessary to develop many utility routines, some of which are included in Fig. 6. Each of these routines was developed in response to a programming problem not covered by the manufacturer's software. The user of minicomputers should expect to have to develop at least some of this kind of software. At the current price levels the manufacturers of minicomputers cannot afford to supply large amounts of special software.

4.3 The Current Software Scene

The software picture is constantly improving as the various minicomputers mature. Most manufacturers engage in a continuous, if not large, program development program. The resulting software usually reflects the needs of a relatively large number of users. Additionally most manufacturers support and encourage the formation of user groups and maintain a users group program library.

At the present time we are investigating the feasibility of changing over from our CPS to DEC's programming system, PS-8. This change would allow us the advantage of DEC's continuous software improvements. Although we feel that the return has made the
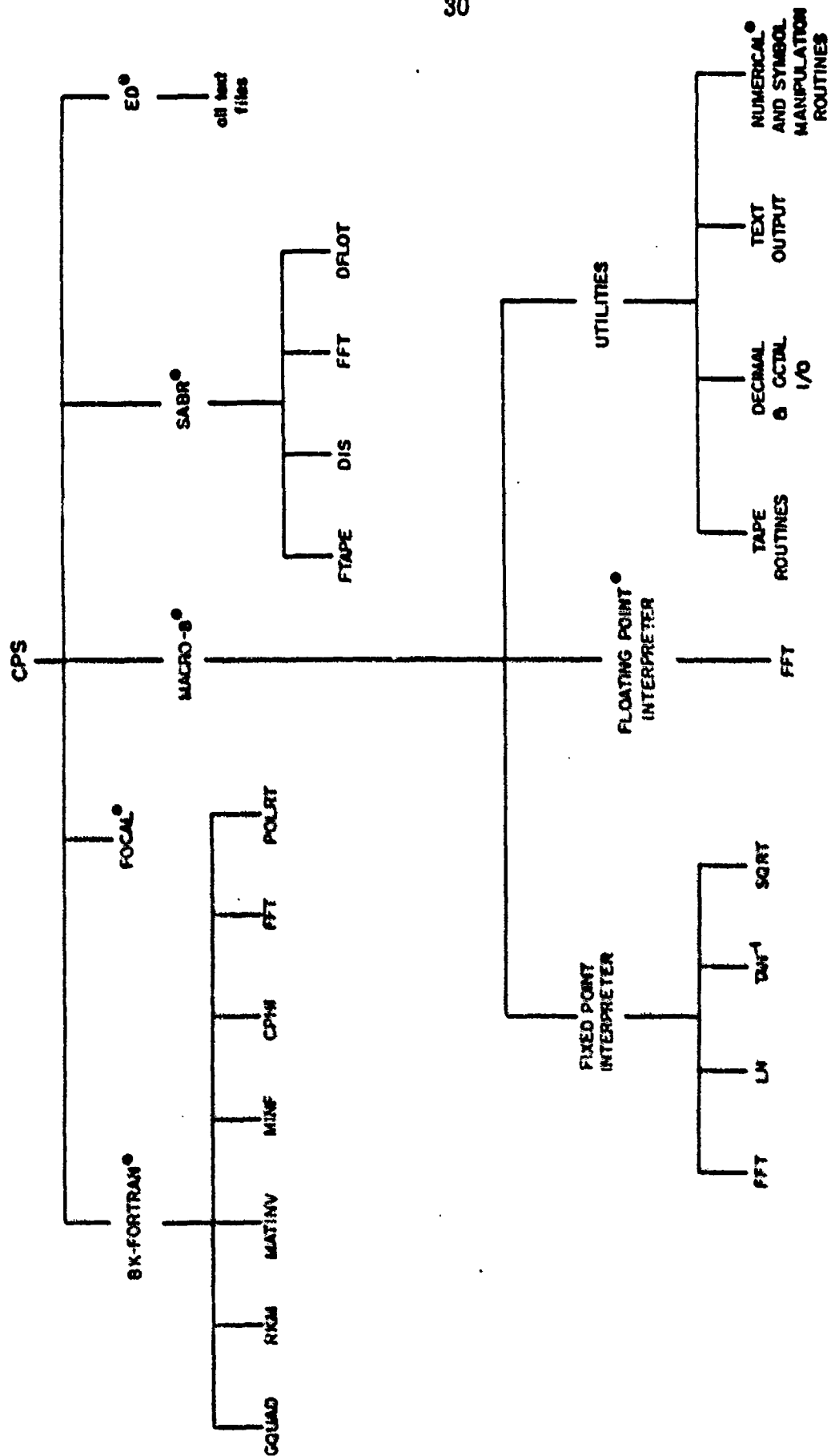
Fig. 6. Programming aids in use at CEL

● DEC SUPPLIED

effort worthwhile, maintaining a system is a large and time-consuming task, and we are constantly investigating the alternatives.

5. CURRENT APPLICATIONS AT CEL

At the present time the CEL computers are being used in the following applications:

> Running propagation experiments
> Post experiment data analysis
> Program development
> Detection theory analysis
> Run-of-the-mill computation
> Classroom demonstrations

5.1 Signal Processing Programs

Our present signal processing programs are of modular construction and are written in non-relocatable assembly language. A foreground-background organization is used with the background (noninterrupt time) being multiprogrammed. These programs are usually about 4000 instructions in length and require about one to two man-months to write. The relatively short development time is due to the frequent use of "canned" support routines.

The support routines currently in vogue are:

> Multiprogrammed monitor (920 locs)
> Double precision fixed point interpreter (384 locs)
> Double precision logarithm and arctan (128 locs)
> Quad precision square root (128 locs)
> Double precision discrete Fourier Transform (512 locs)

Not all of these routines may be included in a particular signal processing program.

5.2 Post Experiment Data Analysis

One of the goals of our signal processing programs is to provide the operator with sufficient on-line information to allow him to evaluate performance and to make any necessary adjustments.

In addition to the processed results presented on-line, most programs have provisions for dumping processed, semi-processed or raw data onto LINC tapes for later analysis. These data dumps are used in attempts to find "hidden" periodicities and relationships. They are also used to extract additional quantitative results and to answer "what if" type questions.

For analyses made in the lab we normally resort to FORTRAN or a mixture of FORTRAN and relocatable assembly language. Here too, we have developed collections of useful routines for performing both computational and I/O chores.

5.3 Program Development

Each signal processing and post experiment analysis program must be written, assembled or compiled, loaded into the computer and tested. Often this process requires a period of time greater than the expected lifetime of the programs being prepared. In order to speed up the process of program development we have invested approximately two man-years in developing and maintaining our own operating systems.

This effort resulted from a two-fold need: first, the software originally supplied with our LINC-8 was inadequate for writing the large programs that we required; and second, we felt that it was necessary to be able to reassemble any processing program in the field (insurance).

DEC now furnishes an operating system (PS-8) for use on PDP-8 computers with 8192 or more words of memory. In many ways the capabilities of PS-8 exceed those of CPS; however, CPS is somewhat more optimized in terms of tape handling and runs faster. More will be said about operating systems in a later lecture.

In addition to our operating systems we have developed many utility packages for use with FORTRAN and assembly language programs.

5.4 Odds and Ends

Once one has a computer and has developed a fairly extensive amount of support software, other applications soon become apparent. Our computers have been used by students and staff for a wide variety of tasks such as:

> Transistor circuit design
> Detection theory calculations
> Phase lock loop studies
> Class work
> Class demonstrations

6. SUMMARY

In general we have found minicomputers to be an effective tool in performing signal analysis and acoustic propagation studies. We are currently using these machines in applications and situations far more complex than we thought possible in 1966.

It is the rare user that finds a manufacturer's software or hardware tailored to his particular requirements. Some points to keep in mind when selecting a minicomputer are:

(1) It probably will be necessary to modify the hardware.

(2) The software will not be quite what you require.

(3) Time spent developing needed programming aids will often pay large dividends.

(4) Organize programs modularly.

(5) Study programs written by others (hopefully by someone more proficient than you are).

(6) Borrow, adapt and modify software wherever useful.

A final point, important enough to merit its own paragraph, is that the hardware designer should not design his hardware in order to ease his work, but to ease the work of the programmer. Additionally, the programmer should not design his software to ease his job, but to ease that of the operator.

35

If a minicomputer is used either in the laboratory, in OEM equipment, on the production line or in the field, it is going to have to be programmed. Selecting and developing the proper software is at least as important as choosing the proper hardware.